

# Optimasi Analisis Sentimen terhadap Kinerja Direktorat Jenderal Pajak Indonesia Melalui Teknik Oversampling dan Seleksi Fitur Particle Swarm Optimization

Nafiatun Sholihah\*<sup>1</sup>, Ferian Fauzi Abdulloh<sup>2</sup>, Majid Rahardi<sup>3</sup>

<sup>1,2,3</sup>Fakultas Ilmu Komputer Universitas AMIKOM Yogyakarta

E-mail: \*[nafiatun@amikom.ac.id](mailto:nafiatun@amikom.ac.id), [ferian@amikom.ac.id](mailto:ferian@amikom.ac.id), [majid@amikom.ac.id](mailto:majid@amikom.ac.id)

## Abstrak

Dalam domain kebijakan publik dan tata kelola pemerintahan, isu perpajakan senantiasa menjadi perhatian khusus di kalangan masyarakat. Dengan tujuan mendapatkan pemahaman yang lebih mendalam tentang pandangan publik terhadap performa Direktorat Jenderal Pajak Indonesia, penelitian ini mengadopsi pendekatan analisis sentimen, menggunakan dataset komentar yang terkumpul dari platform media sosial YouTube. Salah satu kendala signifikan yang dihadapi dalam analisis ini adalah ketidakseimbangan data sentimen komentar, dengan dominasi sentimen positif atau negatif. Dengan demikian, kami menerapkan teknik SMOTE oversampling dan Particle Swarm Optimization (PSO) sebagai strategi seleksi fitur, sebagai bagian dari upaya meningkatkan kualitas model analisis sentimen. SMOTE akan membuat data sintetis dari kelas minoritas sehingga data train akan berimbang dan tidak menghasilkan model yang mengandung bias yang disebabkan ketidak seimbangan data. Selanjutnya dilakukan pemilihan fitur yang dianggap memuat informasi penting untuk meningkatkan performa dari suatu model.

Metode ini terbukti efektif, khususnya pada skenario dengan pembagian data latih sebanyak 70%. Di sini, nilai recall meningkat dari 0.47 menjadi 0.52, sebuah peningkatan yang signifikan dalam mendeteksi sentimen minoritas yang seringkali terabaikan dalam studi sejenis. Selain itu, teknik seleksi fitur menggunakan PSO, dengan menggunakan nilai F1 sebagai kriteria pbest, menghasilkan peningkatan substansial pada semua metrik evaluasi: akurasi mencapai 0.93, recall 0.63, presisi 0.70, dan F1 score 0.66. Ini menunjukkan keefektifan metode tersebut dalam memodelkan berbagai aspek sentimen terhadap perpajakan di Indonesia.

**Kata Kunci**—pajak, sentimen, analisis, pso, smote

## 1. PENDAHULUAN

Perpajakan adalah salah satu pilar utama dalam perekonomian dan tata kelola sebuah negara, termasuk Indonesia. Kinerja Direktorat Jenderal Pajak (DJP) sangat mempengaruhi kepercayaan masyarakat serta efektivitas penerimaan negara[1]. Dalam era digital, persepsi publik seringkali tercermin melalui komentar di media sosial, termasuk platform YouTube, yang menjadi sumber data penting untuk analisis sentimen. Menganalisis sentimen publik ini penting untuk menilai efektivitas kebijakan pajak dan menemukan ruang untuk peningkatan[2].

Namun, analisis sentimen menghadapi tantangan signifikan, yaitu ketidakseimbangan data. Dalam banyak kasus, komentar publik cenderung negatif atau positif secara ekstrim, yang menyebabkan model analisis sentimen bias dan kurang akurat[3]. Tantangan ini menjadi lebih kompleks karena sifat dari data teks yang bersifat high-dimensional dan memiliki struktur yang kurang teratur dibandingkan dengan data numerik[4].

Dalam konteks ini, penelitian ini bertujuan untuk meningkatkan kualitas model analisis sentimen dengan menggunakan teknik oversampling dan Particle Swarm Optimization (PSO) untuk seleksi fitur. Metode ini dipilih berdasarkan temuan penelitian sebelumnya yang

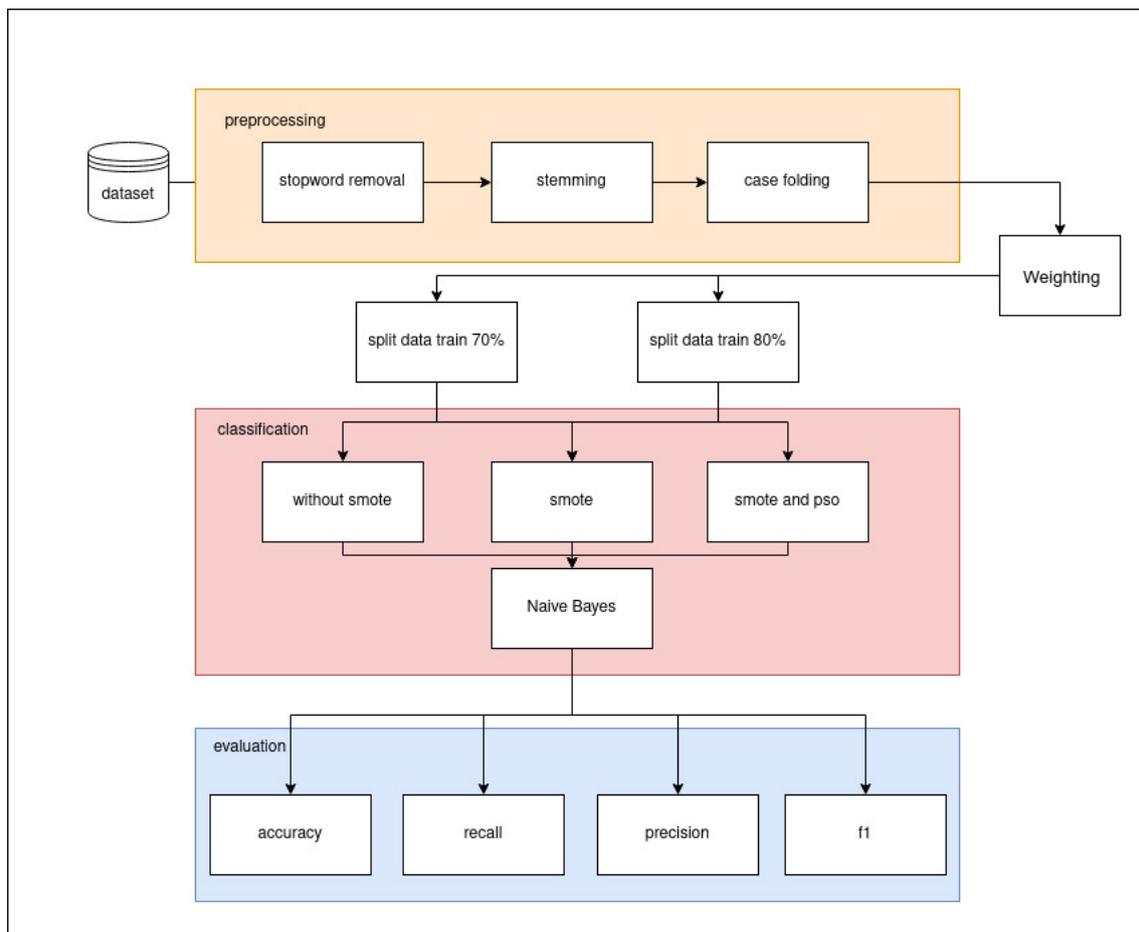
menunjukkan efektivitasnya dalam mengatasi masalah ketidakseimbangan data dan meningkatkan performa model[5].

Secara spesifik, studi ini fokus pada dua skenario berbeda berdasarkan pembagian data pelatihan: 70% dan 80%. Penelitian ini akan mengukur sejauh mana metode yang digunakan dapat meningkatkan metrik evaluasi utama seperti akurasi, recall, presisi, dan F1 score. Selain itu, kami akan mengevaluasi potensi bias model dalam berbagai skenario untuk memastikan bahwa temuan kami dapat diaplikasikan dengan andal dalam praktek.

Melalui pendekatan ini, penelitian ini berupaya memberikan wawasan yang lebih mendalam tentang sentimen publik terhadap kinerja DJP Indonesia. Hasil dari analisis ini diharapkan dapat membantu pembuat kebijakan dalam memahami dan menangani permasalahan atau kebutuhan spesifik terkait perpajakan, serta menawarkan strategi untuk meningkatkan kebijakan dan komunikasi di masa depan..

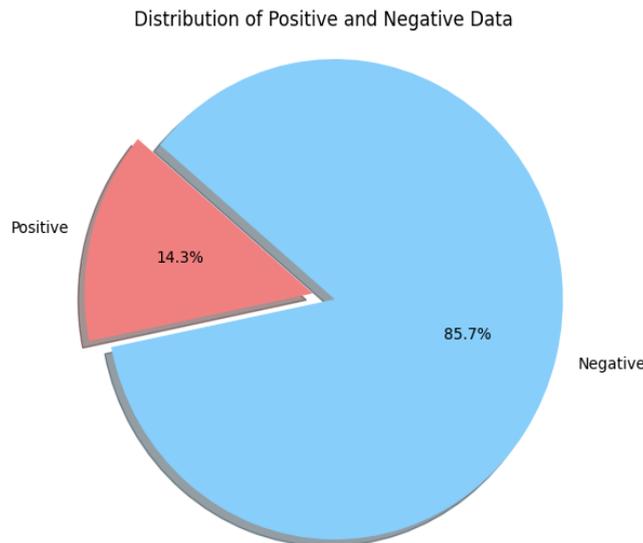
## 2. METODE PENELITIAN

Penelitian ini terdiri atas beberapa tahap, dimulai dari pengumpulan dataset, preprocessing hingga tahap klasifikasi, tahapan penelitian disajikan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Dalam upaya memahami persepsi publik terhadap perpajakan di Indonesia, penelitian ini menggunakan dataset yang dikumpulkan dari komentar pada platform YouTube. Dataset ini didapatkan melalui pencarian dengan keyword "perpajakan Indonesia," sebuah metode yang memungkinkan kami untuk fokus pada diskusi yang secara spesifik berkaitan dengan topik perpajakan di konteks nasional. Menariknya, dataset yang dihasilkan menunjukkan ketidakseimbangan yang signifikan antara komentar positif dan negatif. Secara lebih spesifik, dataset mencakup 84 komentar yang mengungkapkan sentimen positif dan 503 komentar yang menunjukkan sentimen negatif sebagaimana yang disajikan dalam gambar 2.



Gambar 2. Sebaran Data

Ketidakeimbangan ini bukan hanya menantang dari segi analisis, tetapi juga meresahkan karena menunjukkan adanya kemungkinan sentimen publik yang mayoritas negatif terhadap perpajakan atau DJP. Ini juga mempengaruhi metode analisis yang harus digunakan, karena ketidakseimbangan jumlah antara komentar positif dan negatif ini dapat menyebabkan bias dalam model analisis sentimen jika tidak ditangani dengan tepat. Dengan demikian, pemilihan metode yang tepat untuk menangani ketidakseimbangan data—seperti teknik oversampling dan seleksi fitur melalui Particle Swarm Optimization (PSO)—menjadi komponen kritical dalam penelitian ini.

### 2.1. Preprocessing

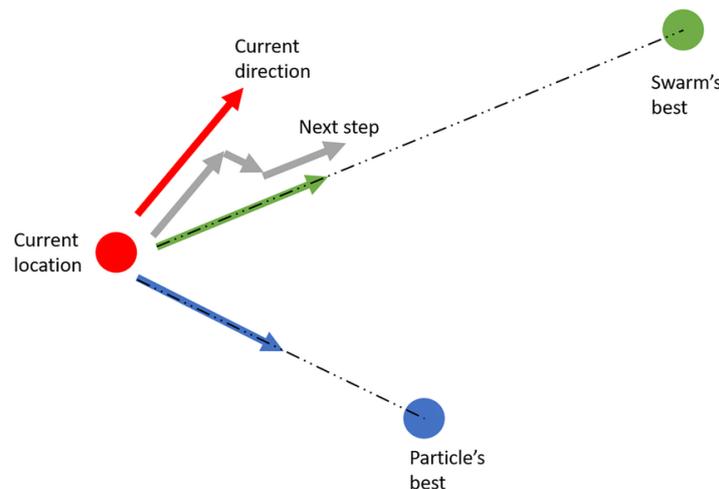
Tahapan preprocessing yang dilakukan meliputi beberapa tahapan, yang pertama adalah stopwords removal untuk menghilangkan kata yang dianggap tidak memiliki bobot, proses ini dilakukan dengan bantuan library nltk pada bahasa pemrograman python, lalu tiap kata akan diubah menjadi kata dasar menggunakan bantuan library sastrawi, lalu semua character non alphanumeric akan dibuang dan semua kata akan diubah menjadi huruf kecil[6].

### 2.2. Pembobotan

Dalam tahap ini tiap komentar akan diubah ke dalam bentuk array/vektor. Vektor tersebut dihitung berdasarkan tiap kata pembentuk komentar. tahap ini dilakukan dengan bantuan modul nltk menggunakan fungsi CountVectorizer[7].

## 2.2. Seleksi Fitur

Particle swarm optimisation atau PSO merupakan algoritma optimisasi yang sering digunakan mudah diimplementasikan, sangat simpel dan umum digunakan. algoritma ini terinspirasi dari perilaku kawanan burung saat mencari makanan. Dalam algoritma PSO, ada tiga variabel yang memiliki peranan penting, yaitu partikel, pbest, dan gbest. Partikel merupakan representasi dari individu yang sedang mencari solusi terbaik. Pbest merujuk kepada posisi terbaik yang pernah dicapai oleh suatu partikel, sementara gbest adalah posisi terbaik dari seluruh partikel dalam kelompok (swarm). Partikel akan mengarah ke arah pbest atau gbest mereka, dengan tujuan akhirnya menemukan posisi gbest terbaik. Untuk memberikan gambaran lebih jelas, konsep ini dapat diilustrasikan melalui Gambar 3. Parameter pbest dan gbest digunakan untuk menghitung kecepatan partikel, yang selanjutnya digunakan untuk menghitung posisi partikel dalam iterasi berikutnya[8].



Gambar 3. Ilustrasi PSO

berikut ini adalah formula matematis dari algoritma pso

$$V_1^1 = WV_1^1 + C_1rand_1 \times (Pbest_1 - X_1^1) + C_2rand_2 \times (Gbest_1 - X_1^1)(1)$$

## 2.3 Oversampling Data

Oversampling data diperlukan apabila data yang kita miliki memiliki label yang tidak seimbang, seperti pada penelitian ini, maka dilakukan pengaplikasian algoritma SMOTE (*Synthetic Minority Oversampling Technique*) untuk mengatasi hal tersebut[9].

Proses SMOTE melibatkan langkah-langkah berikut:

1. Memilih satu sampel acak dari kelas minoritas sebagai titik referensi.
2. Memilih beberapa tetangga terdekat dari titik referensi (biasanya satu atau lebih) dari kelas minoritas yang sama.
3. Menggabungkan titik referensi dengan salah satu tetangga yang dipilih secara acak untuk membuat sampel sintetis. Ini dilakukan dengan mengambil jarak antara titik referensi dan tetangga yang dipilih, mengalikannya dengan angka acak antara 0 dan 1, dan menambahkannya ke koordinat titik referensi. Hasilnya adalah sampel sintetis yang baru.

## 2.4 Klasifikasi

Klasifikasi adalah proses menghasilkan model untuk memprediksi kelas atau kategori[10]. beberapa diantara algoritma klasifikasi yang umum digunakan adalah naive bayes. jenis algoritma naive bayes yang digunakan dalam penelitian ini adalah algoritma multinomial naive bayes yang banyak digunakan dalam analisis text. dalam penelitian ini dataset akan dipecah menjadi 70% data train dan 30% data test. dan juga dilakukan klasifikasi dengan skenario lain dengan memecah dataset menjadi 80% data train dan 20% data test.

## 2.5 Evaluasi

Evaluasi perlu dilakukan untuk mengetahui tingkat keberhasilan dari suatu model. Dalam penelitian ini tingkat keberhasilan ditinjau dari nilai akurasi, recall, precision dan f1 score. penelitian ini akan berfokus pada perbaikan nilai recall, nilai recall digunakan karena recall merupakan nilai banyaknya data dengan label positif yang terdeteksi benar, yang mana dalam penelitian ini data dengan sentiment positive merupakan data minoritas. nilai akurasi yang tinggi namun memiliki recall yang rendah artinya model yang dihasilkan tidak reliable karna model tersebut menghasilkan klasifikasi yang condong ke data mayoritas, untuk itu nilai recall perlu diperhatikan.

Formula akurasi recall precision dan f1 score disajikan dalam formula berikut

$$precision = \frac{TP}{TP+FP} \quad (2)$$

$$recall = \frac{TP}{TP+FN} \quad (3)$$

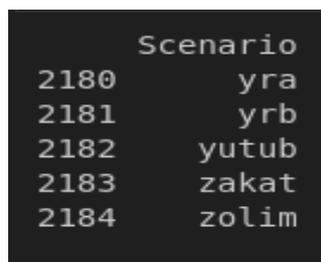
$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (4)$$

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (5)$$

## 3. HASIL DAN PEMBAHASAN

### 3.1. Pembahasan

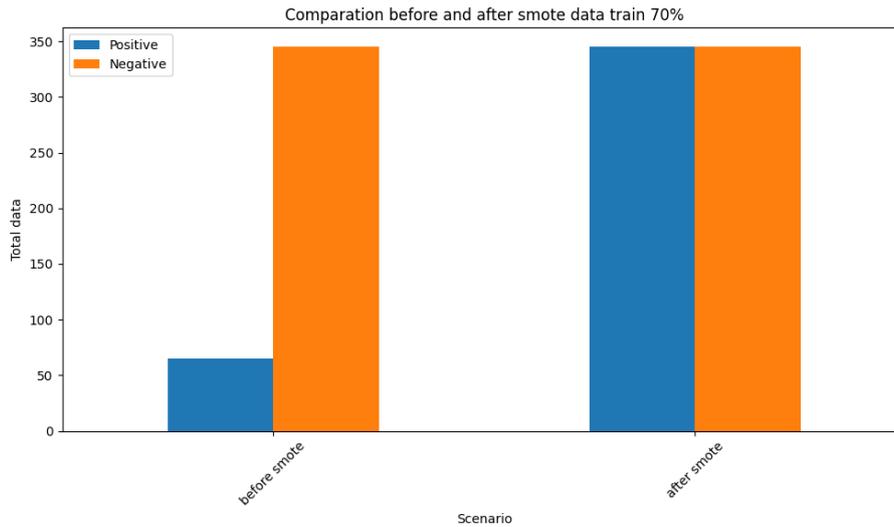
Tahapan penelitian yang dilakukan terdiri atas beberapa proses, pertama-tama akan dilakukan preprocessing dengan bantuan library dari Python3. tahapan preprocessing dijelaskan dalam gambar 1. dalam tahapan preprocessing dihasilkan feature text dengan jumlah 2185 feature, feature yang didapatkan dari 5 video sebagaimana yang ditampilkan pada gambar 4.



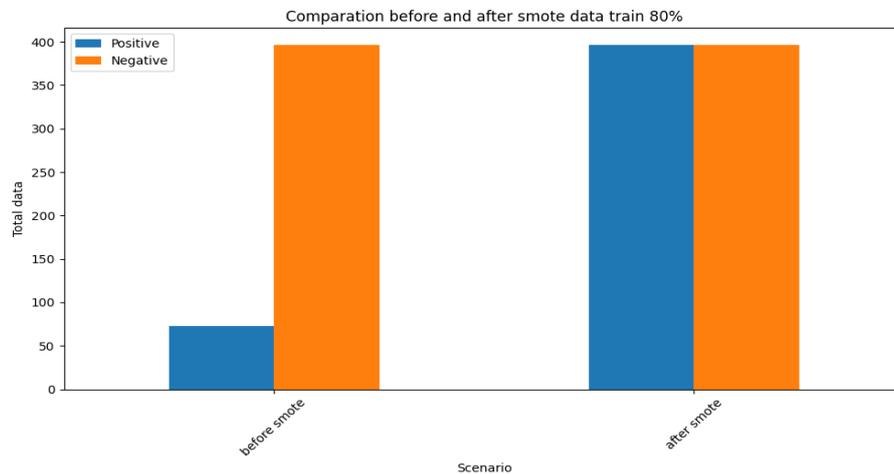
Count	Feature Name
2180	yra
2181	yrb
2182	yutub
2183	zakat
2184	zolim

Gambar 4. Feature dataset

Selanjutnya dilakukan pembobotan dengan menggunakan bantuan library nltk dengan fungsi CountVectorizer untuk mendapatkan matrix dari tiap tiap komentar, lalu pada data matrix tersebut akan dilakukan beberapa skenario, yaitu membagi dataset menjadi 70:30 data train dan dataset dan juga 80:20, lalu juga dilakukan penambahan untuk tiap tiap data train dengan data sintetis yang didapatkan dengan mengaplikasikan algoritma SMOTE, perbedaan data train sebelum dan setelah diaplikasikan algoritma smote disajikan pada gambar 5 dan gambar 6. Setelah itu ke 4 data train akan diaplikasikan kedalam algoritma Naive Bayes dan diukur recall serta akurasi. Hasil dari ke 4 skenario tersebut ditampilkan dalam tabel 1



Gambar 5. Perbandingan Sebelum Dan Setelah Smote 70% dataset



Gambar 6. Perbandingan Sebelum Dan Setelah Smote 80% dataset

oversampling algorithm	data train proportion	accuracy	recall	precision	f1
without smote	70%	0.926554	0.473684	0.750000	0.580645
without smote	80%	0.949153	0.545455	0.666667	0.666667
smote	70%	0.892655	0.526316	0.500000	0.512821
smote	80%	0.923729	0.545455	0.600000	0.571429

Tabel 1. Hasil Evaluasi dengan dan tanpa SMOTE

Hasil yang telah diperoleh pada Tabel 1 menunjukkan beberapa dinamika menarik mengenai efek oversampling dan proporsi data pelatihan terhadap performa model analisis sentimen.

#### Tanpa SMOTE:

Pada proporsi data pelatihan sebesar 70%, model menunjukkan akurasi yang tinggi yaitu 0.926554, tetapi nilai recall cukup rendah (0.473684). Ini menunjukkan bahwa meski model secara umum akurat, ia kurang efisien dalam mengidentifikasi kelas minoritas.

Saat proporsi data pelatihan meningkat menjadi 80%, akurasi juga meningkat menjadi 0.949153. Namun, nilai F1 dan presisi menunjukkan adanya penurunan dibandingkan dengan proporsi data 70%, menandakan potensi bias atau overfitting.

#### Dengan SMOTE:

Pada proporsi data pelatihan 70%, model memiliki akurasi 0.892655 dan recall 0.526316. Meski akurasi lebih rendah dibandingkan tanpa SMOTE, terjadi peningkatan pada recall, yang berarti model lebih baik dalam mengidentifikasi kelas minoritas.

Proporsi data pelatihan 80% menunjukkan peningkatan pada semua metrik evaluasi kecuali akurasi, yang sedikit lebih rendah (0.923729) dibandingkan dengan tanpa SMOTE.

Secara umum, penggunaan SMOTE memperlihatkan peningkatan pada kemampuan model dalam mendeteksi kelas minoritas, seperti yang ditunjukkan oleh nilai recall. Namun, perlu diingat bahwa efektivitas metode ini beragam tergantung pada proporsi data pelatihan. Pada proporsi 70%, SMOTE tampaknya lebih efektif dalam meningkatkan kualitas model dibandingkan dengan proporsi 80%.

Selanjutnya diaplikasikan algoritma *feature selection* PSO kedalam dataset yang telah diseimbangkan komposisi sentimennya menggunakan algoritma SMOTE, dalam algoritma PSO nilai f1 sebagai nilai pbest. Dari skenario ini didapatkan fitur yang dianggap penting dalam pembuatan model, potongan data yang telah terseleksi ditampilkan pada gambar 7

```

Scenario  feature selected
2180     yra              1
2181     yrb              1
2182     yutub             0
2183     zakat             0
2184     zolim             1
    
```

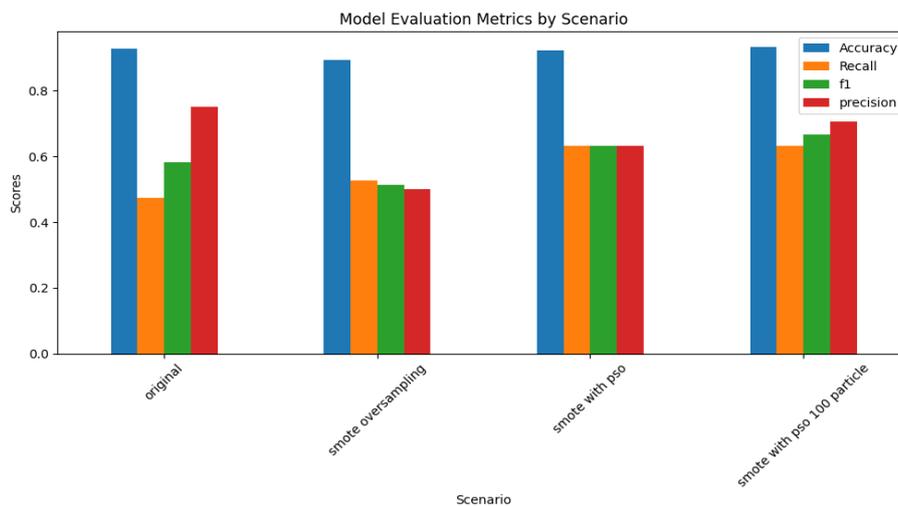
Gambar 7. Feature yang diambil dan dibuang

Pada gambar 7 menjelaskan bahwa kata yang memiliki nilai feature selected 0 akan dibuang pada saat pembuatan model, dalam penelitian ini dilakukan 2 pendekatan, yang pertama dengan menggunakan 50 particle swarm dan 100 particle swarm yang hasil pengujiannya ditampilkan pada tabel 2.

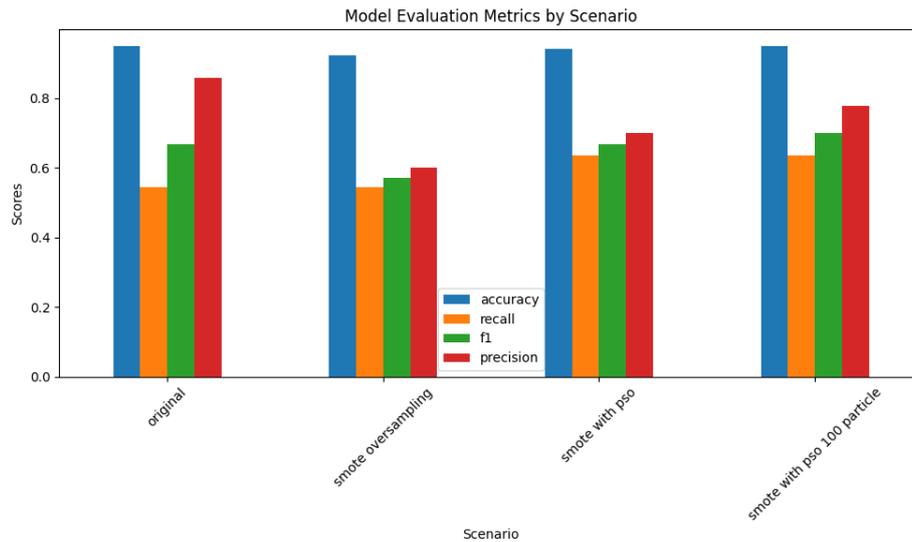
particle	data train proportion	accuracy	recall	precision	f1
50	70%	0.920904	0.631579	0.631579	0.631579
50	80%	0.940678	0.636364	0.700000	0.666667
100	70%	0.932203	0.631579	0.705882	0.666667
100	80%	0.949153	0.636364	0.777778	0.700000

Tabel 2. Hasil Optimasi dengan PSO

Berikut ini juga kami tampilkan seluruh hasil dari penelitian ini dalam bentuk grafik yang kami sajikan pada gambar 8 dan gambar 9



Gambar 8. Hasil Penelitian Pembagian data train 70:30



Gambar 9. Hasil Penelitian Pembagian data train 80:20

Hasil yang diperoleh pada Tabel 2, serta grafik di gambar 8 dan 9 setelah mengimplementasikan algoritma Particle Swarm Optimization (PSO) menunjukkan dinamika yang berbeda dan umumnya lebih positif dalam performa model.

Dengan 50 Partikel:

Pada proporsi data pelatihan 70%, semua metrik (akurasi, recall, presisi, dan F1) menunjukkan peningkatan yang signifikan. Recall dan F1 mencapai 0.631579, menunjukkan model lebih efisien dalam mendeteksi kelas minoritas.

Pada proporsi data pelatihan 80%, terjadi peningkatan pada semua metrik kecuali akurasi. F1 score naik menjadi 0.666667, dan presisi meningkat menjadi 0.700000.

Dengan 100 Partikel:

Pada proporsi data pelatihan 70%, semua metrik evaluasi menunjukkan peningkatan. Akurasi naik menjadi 0.932203, dan F1 meningkat menjadi 0.666667.

Pada proporsi data pelatihan 80%, semua metrik evaluasi juga meningkat. Yang paling mencolok adalah presisi yang mencapai 0.777778 dan F1 yang naik menjadi 0.700000.

Dalam keseluruhan, penambahan algoritma PSO dengan jumlah partikel baik 50 maupun 100 menunjukkan peningkatan performa pada hampir semua metrik evaluasi. PSO terutama memberikan dampak positif terhadap recall dan F1 score, yang penting untuk klasifikasi kelas minoritas. Peningkatan ini lebih konsisten pada proporsi data pelatihan 80%, menunjukkan bahwa PSO mungkin lebih efektif pada dataset yang lebih besar.

## 4. KESIMPULAN

Dengan demikian, kesimpulan utama adalah bahwa penggunaan SMOTE (dengan atau tanpa PSO) dapat meningkatkan kualitas model dengan pembagian data train sebanyak 70%. nilai recall meningkat dari 0.47 menjadi 0.52 yang menunjukkan kemampuan model dalam mendeteksi kelas minoritas meningkat. Sedangkan pada pembagian data train sebanyak 80% terjadi penurunan pada nilai recall f1, dan precision meskipun akurasi naik hal ini menandakan adanya bias model pada skenario ini. Juga dapat disimpulkan pengaplikasian algoritma feature selection particle swarm optimization dengan acuan nilai f1 sebagai pbest dapat meningkatkan seluruh variable evaluasi seperti yang dapat dilihat pada pembagian data train 70% hasil akhir akurasi, recall, presisi, dan f1 berturut turut adalah 0.93, 0.63, 0.70 dan 0.66 sedangkan pada pembagian data train sebesar 80% nilai akurasi, recall, presisi, dan f1 berturut turut adalah 0.94, 0.63, 0.77, dan 0.70.

## DAFTAR PUSTAKA

- [1] Y. M. Rusli, D. P. Nainggolan, and J. A. Fakultas, "PENTINGNYA PENGETAHUAN PAJAK DAN SOSIALISASI PAJAK KEPADA CALON WAJIB PAJAK MASA DEPAN," *J. Pengabd. dan Kewirausahaan*, vol. 5, no. 2, Sep. 2021, doi: 10.30813/JPK.V5I2.2989.
- [2] S. Hikmawan *et al.*, "Sentimen Analisis Publik Terhadap Joko Widodo terhadap wabah Covid-19 menggunakan Metode Machine Learning," *J. Kaji. Ilm.*, vol. 20, no. 2, pp. 167–176, May 2020, doi: 10.31599/JKI.V20I2.117.
- [3] M. Khushi *et al.*, "A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data," *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: 10.1109/ACCESS.2021.3102399.
- [4] M. Sulistiyono, Y. Pristyanto, S. Adi, and G. Gumelar, "Implementasi Algoritma Synthetic Minority Over-Sampling Technique untuk Menangani Ketidakseimbangan Kelas pada Dataset Klasifikasi," *Sist. J. Sist. Inf.*, vol. 10, no. 2, pp. 445–459, May 2021, doi: 10.32520/STMSI.V10I2.1303.
- [5] S. Pramukti, S. D. Pramukti, A. Nugroho, and A. S. Sunge, "Analisis Sentimen Masyarakat Dengan Metode Naïve Bayes dan Particle Swarm Optimization," *Techno.Com*, vol. 21, no. 1, pp. 61–74, Feb. 2022, doi: 10.33633/tc.v21i1.5332.
- [6] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment Analysis Based on Deep Learning: A Comparative Study," *Electron. 2020, Vol. 9, Page 483*, vol. 9, no. 3, p. 483, Mar. 2020, doi: 10.3390/ELECTRONICS9030483.
- [7] H. Jelodar *et al.*, "A NLP framework based on meaningful latent-topic detection and sentiment analysis via fuzzy lattice reasoning on youtube comments," *Multimed. Tools Appl.*, vol. 80, no. 3, pp. 4155–4181, 2021, doi: 10.1007/s11042-020-09755-z.
- [8] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proc. ICNN'95 - Int. Conf. Neural Networks*, vol. 4, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.
- [9] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002.
- [10] B. Charbuty and A. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning," *J. Appl. Sci. Technol. Trends*, vol. 2, no. 01, pp. 20–28, Mar. 2021, doi: 10.38094/jastt20165.